

ABSTRACT OF THE DISCLOSURE

Multiple instructions, specifying equivalent operations but designating different execution units, are stored beforehand on an instruction exchange table. First, a primary compiler compiles a source program into a set of machine-readable instructions. From the set of instructions, an instruction parallelizer generates a set of long instruction words. Specifically, an instruction identifier identifies one of the instructions in the set with one of the instructions stored on the instruction exchange table. Then, an instruction replacer replaces the instruction in question with another one of the instructions that is also stored on the instruction exchange table, specifies an equivalent operation but designates a different execution unit as a target. In this manner, the number of parallelly executable instructions can be increased, while the number of no-operation instructions can be reduced, thus generating a parallelized instruction set at a higher level of parallelism.